PATENT
ATTORNEY DOCKET NO.: 111788.00101

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of	)	
	)	
William M. JOHNS et al.	)	
	)	
Serial No. 09/765,431	)	
	)	Group Art Unit: 2152
Confirmation No.: 9036	)	
	)	Examiner: Lan Dai T. Truong
Filed: January 22, 2001	)	
	)	
For: SYSTEM AND METHOD FOR CONTINUOUS MONITORING AND MEASUREMENT OF PERFORMANCE OF COMPUTERS ON NETWORK	)	

APPEAL BRIEF

Mail Stop Appeal Brief – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

The Appellants, through undersigned counsel, respectfully submit the present Appeal Brief in support of the Notice of Appeal filed February 8, 2008, and in response to the Notice of Non-Compliant Appeal Brief mailed November 14, 2008.

I. Real party in interest

The real party in interest is the assignee, InsightETE Corporation.

II. Related appeals and interferences

There are no related appeals or interferences.

III. Status of claims

Claims 1-14 are pending in the application, stand finally rejected, and form the subject matter of the present appeal.

IV. Status of amendments

An Amendment after Final Rejection, amending only the drawings, is filed concurrently herewith. No amendment to the claims has been filed since the Final Rejection mailed November 8, 2007.

V. Summary of claimed subject matter

The invention defined by claim 1 and the claims dependent therefrom is directed to a method for monitoring performance and availability of application servers on a network, including a percentage of time that each of the application servers is available to an end user relative to the time the application servers are intended to be available and a responsiveness of the application servers to the end user in terms of a delay between the end user's entering data into a workstation keyboard and a response from one of the application servers with new data on the user's workstation screen, the method comprising:

(a) running at least one performance monitor process (Fig. 1, performance monitors 1-13a,b,c; page 4, line 25,through page 5, line 2 (sending pseudo messages routed from application server to user location and tracking message's time, i.e., watching network activity to and from the application servers to entry servers which connect the network to the end user's workstation); page 10, line 12, through page 11, line 7) on the network, said at least one

performance monitor process watching network activity to and from the application servers (Fig. 2, application servers 2-20; page 12, line 19) to entry servers which connect the network to the end user's workstation (Fig. 2, users 2-8; page 4, user workstations in line 1) and creating a transaction response time log and activity audit trail for the network (Fig. 1, Log File 1-5; page 12, lines 5-7, "A Log File 1-5 is created that summarizes all SNMP alerts received and is used for auditing and insuring that all alerts were properly handled.");

(b) running a network monitor manager process on the network, for consolidating information from the transaction response time log (Fig. 1, BDNetManager 1-1; page 10, line 20, through page 11, line 7);

(c) establishing a connection from the network monitor manager process to said at least one performance monitor process to control said at least one performance monitor to send a pseudo message for tracking time in the network to an entry server to determine said network availability (Fig. 1, connection shown in dashed line between network monitor (BDNetManager) 1-1 and network monitor (BDNetMon) 1-19; page 10, lines 21-24); and

(d) receiving the pseudo message from said at least one performance monitor process and determining a response for the pseudo message for each segment of the network traversed by the pseudo message to determine where problems regarding said availability exist within the network connection for the entry server (Fig. 1, Monitor_Log 1-18 and Avail_Log 1-8; page 10, line 25, through page 11, line 7 and page 11, lines 16-23 more particularly describing the determination of system availability from an end user's perspective).

Claim 3 depends from claim 1 and adds the further limitations:

(i) running a client-server monitoring process on a server dedicated to the client-server monitoring process (Fig. 2, Client/Server Monitoring System (CSMS) 2; page 12, lines 13-20);

(j) receiving, in the client-server monitoring process, information about transactions executed by production applications on the network (Fig. 2, Filtering Agents 2-6, 2-19, 2-25; page 12, lines 23-25); and

(k) determining performance and availability of the production applications in accordance with the information received in step (j) (Fig. 2, BDManager 2-12; page 14, lines 3-7).

Claim 4 depends from claim 3 and adds the further limitation that step (j) comprises running a filtering agent on each or on behalf of each of the production applications to convert the information from application logs into a form usable by the client-server monitoring process (Fig. 2, Filtering Agents 2-6, 2-19, 2-25; page 13, lines 2-6).

Claim 5 depends from claim 4 and adds the following further limitations:

the network comprises a mainframe having at least one logical partition which generates an application log (Fig. 3, mainframe 3-1; page 16, lines 9-11); and

the method further comprises (1) monitoring the application log through a mainframe monitoring process (Fig. 3, BDMVSFilter 3-13; page 16, lines 13-18).

Claim 6 depends from claim 5 and adds the following further limitations:

the application log comprises transaction entries having end-user addresses; and

step (l) comprises categorizing the transaction entries by the end-user addresses (page 16, lines 20-23).

Claim 12 depends from claim 4 and adds the further limitations that each said filtering agent detects processes running on the network and cross-references the detected processes to known processes, and further comprising forming an event correlation engine in accordance with the detected processes (page 16, lines 3-7).

Claim 13 depends from claim 12 and adds the further limitations that each said filtering agent detects changes to the processes running on the network, and further comprising maintaining the event correlation engine in accordance with the detected changes to the processes (page 16, lines 3-7).

Claim 14 depends from claim 13 and adds the further limitation of, when it is determined in step (k) that the performance or the availability of one of the production applications is impaired, determining and reporting a cause of impairment and its corresponding effect on a service level agreement (SLA) in accordance with the event correlation engine (Fig. 3, SLAByLocation table 4-6 and SLAProcess table 4-3; page 17, lines 1-8).

VI. Grounds of rejection to be reviewed on appeal

A. The rejection of claims 1-3 under 35 U.S.C. § 103(a) over *Curley et al* in view of *Phaal*

B. The rejection of claims 4-11 under 35 U.S.C. § 103(a) over *Curley et al* in view of *Phaal* and further in view of *Gretta, Jr.*

C. The rejection of claims 12 and 13 under 35 U.S.C. § 103(a) over *Curley et al* in view of *Phaal* and *Gretta, Jr.*, and further in view of *Goldsack et al*

D. The rejection of claim 14 under 35 U.S.C. § 103(a) over *Curley et al* in view of *Phaal, Gretta, Jr.*, and *Goldsack et al* and further in view of *Chen et al*

VII. Argument

A. The rejection of claims 1-3 under 35 U.S.C. § 103(a) over *Curley et al* in view of *Phaal*

The Appellants respectfully urge reversal of the rejection of claims 1-3 under 35 U.S.C. § 103(a) over *Curley et al* in view of *Phaal*. Unlike *Curley et al,* the present claimed invention is not limited to HTTP, but can instead be used with a wide variety of other protocols.

The Final Rejection acknowledges that *Curley et al* does not teach each segment of the network being monitored and determining network problems, but instead cites *Phaal* for that teaching. However, since *Curley et al* teaches a Web server, in which communication propagates over the worldwide Internet between servers and clients, often along unpredictable paths, it would have been unrealistic and therefore non-obvious to monitor each segment of the network within the context of *Curley et al,* with or without the further teachings of *Phaal.* Therefore, a person having ordinary skill in the art who had reviewed the applied references would not have found the present claimed invention obvious.

B. The rejection of claims 4-11 under 35 U.S.C. § 103(a) over *Curley et al* in view of *Phaal* and further in view of *Gretta, Jr.*

1. In general and claim 4

Even if the rejection of claims 1-3 is affirmed, the Appellants respectfully urge reversal of the rejection of claims 4-11. These claims recite running a filtering agent on each or on behalf of each of the production applications to convert the information from application logs into a form usable by the client-server monitoring process. In other words, it is information *from application logs* that is converted into a form usable by the client-server monitoring process. This is a form of data normalization.

The Final Rejection acknowledges that neither *Curley* nor *Phaal* teaches or suggests any such thing, but instead cites *Gretta, Jr.* for that teaching. However, what *Gretta, Jr.* actually teaches is filtering *packets,* not information from application logs, so that they can be monitored, which is not at all the same thing as what is recited in present claims 4-11. Therefore, the asserted combination of references would not have taught, suggested, resulted in, or otherwise rendered obvious the subject matter of present claims 4-11.

2. Claim 5

Even if the rejection of claim 4 is affirmed, the Appellants respectfully urge reversal of the rejection of claim 5. This claim recites that the network comprises a mainframe having at least one logical partition which generates an application log and that the method further comprises monitoring the application log through a mainframe monitoring process.

None of the applied references teaches or suggests a mainframe having at least one logical partition which generates an application log. In fact, *Curley et al* is concerned with an HTTP server, which is different from a mainframe and in fact a mutually exclusive category of

device. None of the paragraphs of *Curley et al* cited in the rejection of claim 5 teach or suggest a mainframe having at least one logical partition which generates an application log.

In rejecting claim 5, the Final Rejection asserts that "Curley discloses method for setting thresholds for monitoring procedures in order to determine network fails/errors...." Even if that characterization of the reference is accepted *arguendo* as true, it still does not meet the above-noted limitations of present claim 5. Therefore, the Appellants respectfully submit that a *prima facie* case of obviousness has not been made out.

3. Claim 6

Even if the rejection of claim 5 is affirmed, the Appellants respectfully urge reversal of the rejection of claim 6. According to claim 6, the application log comprises transaction entries having end-user addresses; and step (l) comprises categorizing the transaction entries by the end-user addresses.

The applied references teach no such thing. The cited portion of *Curley et al* teaches monitoring transactions involving a network address, but does not teach or suggest categorizing the transaction entries by the end-user addresses. Therefore, the combination of references asserted against claim 6 would not have taught, suggested, resulted in, or otherwise rendered obvious the subject matter of claim 6.

C. The rejection of claims 12 and 13 under 35 U.S.C. § 103(a) over *Curley et al* in view of *Phaal* and *Gretta, Jr.*, and further in view of *Goldsack et al*

Even if the previous grounds of rejection are affirmed, the Appellants respectfully urge reversal of the rejection of claims 12 and 13. The Final Rejection acknowledges that none of

Curley, Phaal and *Gretta* teaches the limitations of claims 12 and 13 regarding an event correlation engine, but instead cites *Goldsack et al* for teaching that "the received data is match to selection criterion in order to determine performance results...." However, the Final Rejection does not indicate where *Goldsack et al* teaches that an *event correlation engine* is formed or maintained as recited in claims 12 and 13. Therefore, the Appellants respectfully submit that a *prima facie* case of obviousness has not been made out with regard to claims 12 and 13.

D. The rejection of claim 14 under 35 U.S.C. § 103(a) over *Curley et al* in view of *Phaal, Gretta, Jr.*, and *Goldsack et al* and further in view of *Chen et al*

Even if the other grounds of rejection are affirmed, the Appellants respectfully urge reversal of the rejection of claim 14. Claim 14 recites, when it is determined in step (k) that the performance or the availability of one of the production applications is impaired, determining and reporting a cause of impairment and its corresponding effect on a service level agreement (SLA) in accordance with the event correlation engine.

The Final Rejection acknowledges that none of *Curley, Phaal, Gretta* and *Goldsack* teaches or suggests any such limitation. Instead, the Final Rejection cites *Chen et al* and asserts that "Chen teaches a real time network monitoring system which is capable to determine network problem and then subsequently provide scheme for repairing the discovered network problem."

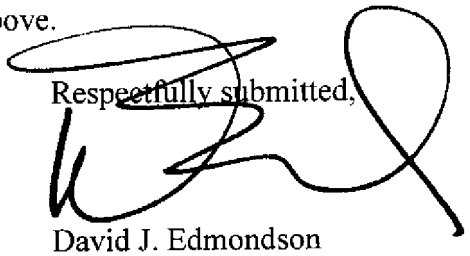
However, claim 14 does not recite using the data for repair. Instead, claim 14 recites determining and reporting a cause of impairment and its corresponding effect on a service level agreement (SLA) in accordance with the event correlation engine, which is quite different.

Therefore, the Appellants respectfully submit that the combination of references asserted against claim 14 would not have taught, suggested, resulted in, or otherwise rendered obvious the subject matter of claim 14.

For the reasons set forth above, the Appellants respectfully urge reversal of the rejection of all pending claims.

Please charge any deficiency in fees, or credit any overpayment thereof, to Deposit Account No. 23-2185 (111788-00101). In the event that a Petition for Extension of Time is required to render the present submission timely and either is not submitted herewith or is insufficient to render the present submission timely, the Applicant hereby petitions under 37 C.F.R. § 1.136(a) for such an extension for as many months as are required to render the present submission timely. Any fee due is authorized above.

Respectfully submitted,



David J. Edmondson
Reg. No. 35,126

BLANK ROME LLP
600 New Hampshire Ave., N.W.,
Washington, D.C. 20037-2403
202-772-5838 (Phone)
202-572-1438 (Facsimile)

VIII. Claims appendix

Following is a list of the claims on appeal in the current form.

1. A method for monitoring performance and availability of application servers on a network, including a percentage of time that each of the application servers is available to an end user relative to the time the application servers are intended to be available and a responsiveness of the application servers to the end user in terms of a delay between the end user's entering data into a workstation keyboard and a response from one of the application servers with new data on the user's workstation screen, the method comprising:

(a) running at least one performance monitor process on the network, said at least one performance monitor process watching network activity to and from the application servers to entry servers which connect the network to the end user's workstation and creating a transaction response time log and activity audit trail for the network;

(b) running a network monitor manager process on the network, for consolidating information from the transaction response time log;

(c) establishing a connection from the network monitor manager process to said at least one performance monitor process to control said at least one performance monitor to send a pseudo message for tracking time in the network to an entry server to determine said network availability; and

(d) receiving the pseudo message from said at least one performance monitor process and determining a response for the pseudo message for each segment of the network traversed by

the pseudo message to determine where problems regarding said availability exist within the network connection for the entry server.

2 The method of claim 1, further comprising:

(e) running at least one availability monitor process on the network;

(f) from the response determined in step (d), detecting at least one possibly failed component of the network;

(g) sending a message from the at least one availability monitor process to the at least one possibly failed component; and

(h) determining, in accordance with a result of the message, whether the at least one possibly failed component has failed.

3. The method of claim 1, further comprising:

(i) running a client-server monitoring process on a server dedicated to the client-server monitoring process;

(j) receiving, in the client-server monitoring process, information about transactions executed by production applications on the network; and

(k) determining performance and availability of the production applications in accordance with the information received in step (j).

4. The method of claim 3, wherein step (j) comprises running a filtering agent on each or on behalf of each of the production applications to convert the information from application logs into a form usable by the client-server monitoring process.

5. The method of claim 4, wherein:

the network comprises a mainframe having at least one logical partition which generates an application log; and

the method further comprises (1) monitoring the application log through a mainframe monitoring process.

6. The method of claim 5, wherein:

the application log comprises transaction entries having end-user addresses; and

step (l) comprises categorizing the transaction entries by the end-user addresses.

7. The method of claim 6, further comprising (m) generating a performance report for the network through an administrative process and making the report available over a data network.

8. The method of claim 7, wherein the data network comprises the Internet.

9. The method of claim 8, further comprising:

(n) receiving, in the client-server monitoring process, information about transactions executed by e-commerce applications on the network; and

(o) determining performance and availability of the e-commerce applications in accordance with the information received in step (n) through an e-commerce monitoring process.

10. The method of claim 9, wherein at least one of the e-commerce applications makes at least one Web page accessible to customers, and wherein step (n) comprises placing code in the at least one Web page, the code sending time stamps to the client-server monitoring process when the code is accessed.

11. The method of claim 10, further comprising providing a central data repository, and wherein the network monitor manager process, the client-server monitoring process, the mainframe monitoring process, the administrative process, and the c-commerce monitoring process communicate with one another through the central data repository.

12. The method of claim 4, wherein each said filtering agent detects processes running on the network and cross-references the detected processes to known processes, and further comprising forming an event correlation engine in accordance with the detected processes.

13. The method of claim 12, wherein each said filtering agent detects changes to the processes running on the network, and further comprising maintaining the event correlation engine in accordance with the detected changes to the processes.

14. The method of claim 13, further comprising, when it is determined in step (k) that the performance or the availability of one of the production applications is impaired, determining and reporting a cause of impairment and its corresponding effect on a service level agreement (SLA) in accordance with the event correlation engine.

IX. Evidence appendix

None

X. Related proceedings appendix

None